CHUYÊN ĐỀ
TỔ CHỨC DỮ LIỆU
CÁC KIỂU DỮ LIỆU NÂNG CAO 1
CON TRỎ VÀ MẢNG

Giảng viên: VŨ QUỐC HOÀNG
(vqhoang@fit.hcmus.edu.vn)

Nội dung trình bày

- Kiểu tham chiếu
- Kiểu con trỏ
- Kiểu mảng
- Kiểu chuỗi
- Kiểu hàm
- Con trỏ và cấp phát động
- Các lỗi thường gặp với con trỏ

Kiểu tham chiếu
Đối tượng và tham chiếu



227 Nguyễn Văn Cừ, Quận 5, TP. HCM

Kiểu tham chiếu
Đối tượng và tham chiếu

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP. HỒ CHÍ MINH KHTN

Tiếng Việt English

ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Thứ Năm, 23/06/2011, 03:07 (GMT+7)

Tìm kiếm

- Trang Chủ
- Giới Thiệu
- Đào tạo
- Tuyển sinh
- Khoa học CN-QHQT
- Công tác Sinh viên
- Tổ chức hành chánh
- Đoàn thể
- Thư viện
- Các Khoa
- Các Phòng Thí Nghiệm
- Các Trung Tâm
- Các phòng ban
- Lịch công tác
- Webmail Cán bộ
- Webmail Sinh viên
- Liên hệ
- Download Logo
- Hình ảnh

Chào mừng các bạn đến với website
Trường ĐH Khoa học Tự nhiên

Giới thiệu

Năm 1996, Trường ĐH KHTN được chính thức thành lập theo quyết định 1236/GDĐT của Bộ GD&ĐT ngày 30/3/1996 trên cơ sở tách ra từ Trường Đại học Tổng hợp TP.HCM để tham gia vào Đại học Quốc gia TP.HCM. Trường hiện có 09 Khoa; 15 trung tâm đào tạo, nghiên cứu khoa học- chuyển giao công nghệ. Hiện nay, Trường đang có 32...

Chi tiết...

Tin Khoa học tự nhiên

ĐH KHTN: Trao bằng Tiến sĩ và Thạc sĩ cho 347 NCS và HVCH

16/06/2011, 10:47 (GMT+7)

Sáng ngày 14, 15/6, trường ĐH KHTN TP.HCM đã tổ chức Lễ trao bằng Tiến sĩ và Thạc sĩ cho 24 nghiên cứu sinh (NCS) và 323 học viên cao học (HVCH) các khóa 15, 16, 17, 18, 19 thuộc các khoa Công nghệ Thông tin, Toán-Tin, Vật lý, Điện tử-Viễn thông, Địa chất, Hóa, Sinh, Môi trường.

...

Chi tiết...

Các hoạt động khác

- ĐH KHTN TP.HCM và IBM Việt Nam hợp tác thành lập "Trung tâm Công nghệ Xuất sắc – CoE IBM-UNS" (10/06/2011)
- ĐH KHTN hợp tác đào tạo cử nhân Quản lý Kinh doanh Quốc tế với ĐH Keuka, Hoa Kỳ (08/06/2011)
- Quy chế đào tạo theo tín chỉ sửa đổi: Giúp sinh viên có thêm cơ hội tốt nghiệp (03/06/2011)
- Những hình ảnh đầu tiên của Đội tuyển ĐH KHTN tại vòng chung kết cuộc thi Lập trình sinh viên Quốc tế ACM/ICPC tại Mỹ (31/05/2011)
- Thực phẩm chức năng hỗ trợ điều trị ung thư (24/05/2011)
- Nghiên cứu trồng rau không cần đất ở Trường Sa (16/05/2011)
- Sinh viên ĐH KHTN phát huy truyền thống nghiên cứu, sáng tạo tại S-Ideas lần IV năm 2011 (09/05/2011)
- Thêm 364 cử nhân ngành CNTT hệ HCĐH ra trường (28/04/2011)
- ĐH KHTN TP.HCM: Gần 1000 sinh viên tham gia ngày hội việc làm ngành CNTT (27/04/2011)

Hệ đào tạo từ xa qua mạng

Đăng nhập

Tên tài khoản

Mật khẩu

Đăng nhập

Lưu thông tin | Quên mật khẩu

Kích hoạt tài khoản | Hướng dẫn

Thông báo đào tạo

- Thông báo v/v nội trú KTX 135B Trần Hưng Đạo cho sinh viên K2008 và K2009 hệ ĐHCQ (22/06/2011) new!
- V/v lấy ý kiến phản hồi từ người học về hoạt động giảng dạy của Nhà trường HK2/10-11 (22/06/2011) new!
- Hình lễ trao bằng Thạc sĩ, Tiến sĩ ngày 14,15/6/2011 (21/06/2011)

http://www.hcmus.edu.vn/

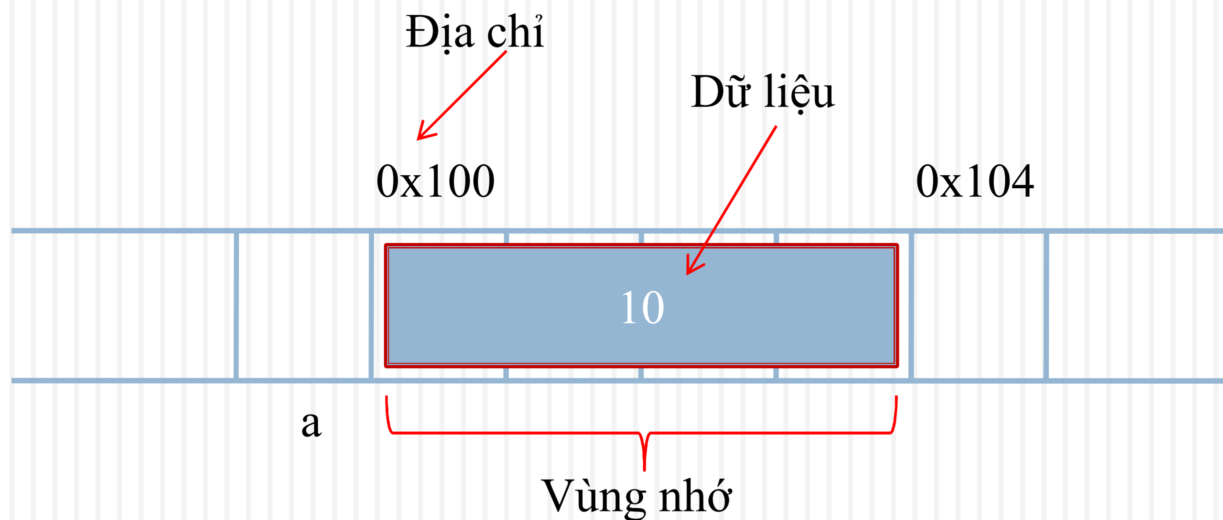
Kiểu tham chiếu
Tham chiếu tuyệt đối và tương đối

- 2/15 Nguyễn Văn Cừ, Q5, TP HCM
  - Tham chiếu tuyệt đối
- Nhà thứ 2 bên trái, trong Hẽm 15 Nguyễn Văn Cừ, Q5, TP HCM
  - Tham chiếu tương đối gồm:
    - Tham chiếu cơ sở: hẽm 15, Q5, TP HCM
    - Độ lệch tương đối: thứ 2, bên trái

Kiểu tham chiếu
Dữ liệu và địa chỉ

- Là một dạng cụ thể của đối tượng/tham chiếu: đối tượng/(*nơi chứa*)/địa chỉ
  - Đối tượng: dữ liệu
  - *Nơi chứa*: vùng nhớ (dãy các ô nhớ)
  - Địa chỉ: con số xác định vùng nhớ (thường là ô nhớ đầu tiên)

```
int a;
a = 10;
```



Kiểu con trỏ

- Là một hệ thống kiểu tham chiếu dựa trên địa chỉ
  - Địa chỉ ảo (địa chỉ logic) trong không gian địa chỉ ảo
  - Một số nguyên không dấu 8/16/32/64 bít tùy nền tảng máy
  - Hằng địa chỉ thường ghi bằng số hexa (cơ số 16)
- Cũng có thể xem là kiểu cơ bản trong C/C++
  - Hỗ trợ từ kiến trúc máy
  - Rất quan trọng và hay dùng để tạo nên các kiểu khác
  - Được trình biên dịch hỗ trợ mạnh

Kiểu con trỏ
Các thao tác cơ bản

- Khai báo

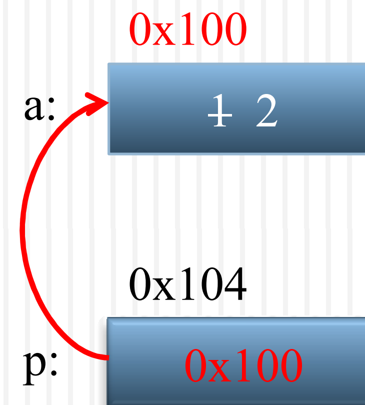
```
int a = 1;
int *p;
```

- Trỏ

```
p = &a; //p=0x100
```

- Truy cập dữ liệu được trỏ

```
*p = *p + a; //*p=a=2
```



Kiểu con trỏ
Một số thao tác khác

- Bỏ trỏ

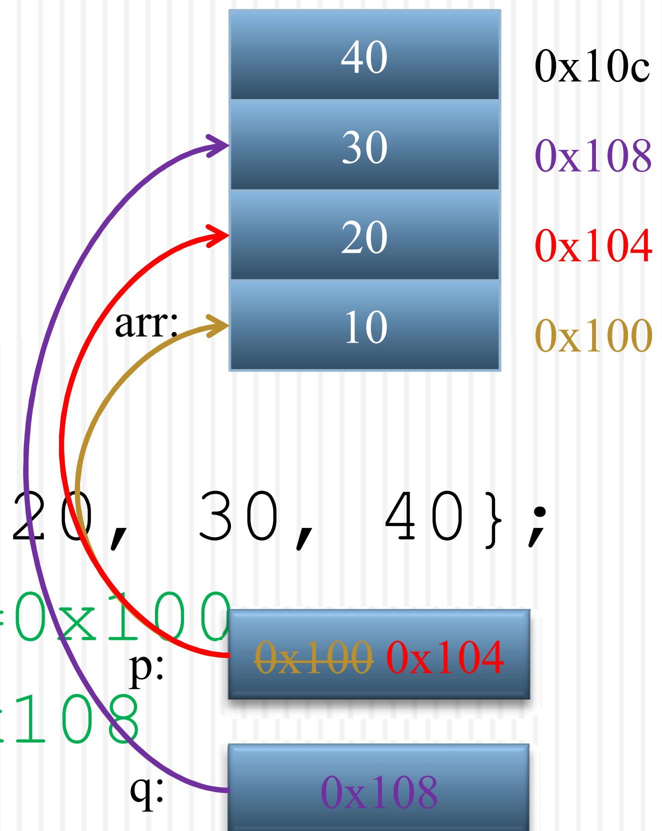
```
p = NULL;
```

- Cùng trỏ

```
int *q = p;
```

- Tăng giảm con trỏ

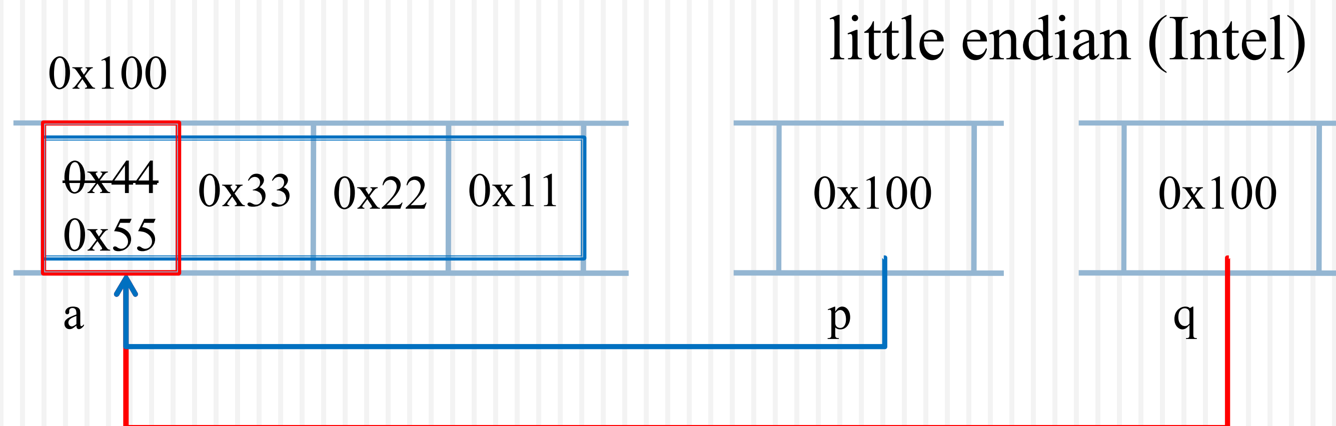
```
int arr[] = {10, 20, 30, 40};
p = &arr[0]; //p=0x100
q = p + 2; //q=0x108
p++; //p=0x104
```



Kiểu con trỏ
Ép kiểu con trỏ

- Kiểu xác định khuôn dạng dữ liệu

```
int a = 0x11223344;
int *p = &a; //p=0x100
char *q = (char*)p; //q=0x100
*q = 0x55; //a=0x11223355
```



Kiểu con trỏ
Tham chiếu của C++

- C++ đưa ra khái niệm tham chiếu
  - Hậu trường: là kiểu con trỏ
  - Cú pháp: giống kiểu dữ liệu bình thường
  - Logic: là nickname (alias)

```
int a = 10;
int *p = &a;
int &b = a; //b cũng như a
*p = 20; //*p=a=b=20
b = 30; //b=a=*p=30
int *q = &b; //q=&b=&a=p
*q = 40; //*q=b=a=*p=40
```

Kiểu con trỏ
Con trỏ và tham số của hàm

- Tham trị (pass by value)

```
void incr1(int i) { i++; }
```

- Tham trỏ (pass by address)

```
void incr2(int *i) { (*i)++; }
```

- Tham chiếu (pass by reference)

```
void incr3(int &i) { i++; }
```

```
int v = 10;
incr1(v); //v=10
incr2(&v); //v=11
incr3(v); //v=12
```

Kiểu con trỏ
Con trỏ cấp cao

- Con trỏ có thể trỏ đến con trỏ

```
    void f1(int v) { v++; }
    void f2(int *v) { (*v)++; v++; }
    void f3(int **v) {(*(*v))++; (*v)++; }
    void f4(int *&v) { (*v)++; v++; }
int arr[] = {10, 20, 30, 40};
int *p = arr; //p=&arr[0]
f1(*p); //p=&arr[0], arr[0]=10
f2(p); //p=&arr[0], arr[0]=11
f3(&p); //p=&arr[1], arr[0]=12
f4(p); //p=&arr[2], arr[1]=21
```

Kiểu mảng

- Kiểu mảng là một kiểu danh sách, quản lý các phần tử cùng kiểu, được lưu trữ với kích thước như nhau và liên tiếp nhau trong bộ nhớ
- Trong C/C++, kiểu mảng là một kiểu tham chiếu tương đối gồm: địa chỉ cơ sở và chỉ số tương đối

```
int arr[] = {10, 20, 30, 40};
arr[2] = arr[0];
int *p = &arr[2];
```

Kiểu mảng

- Mảng trong C/C++ là hằng con trỏ

```
p = arr + 2; //p=&arr[2]
arr++; //error
arr[3] = *(arr + 1); //arr[3]=arr[1]
p[0] = p[1]; //*p=*(p+1)
```

- Mảng trong C/C++ không lưu trữ tường minh kích thước mảng (số lượng phần tử)

```
int arr[] = {10, 20, 30, 40};
arr[4] = 50; //!!!
```

Kiểu chuỗi

- Nhiều ngôn ngữ hỗ trợ kiểu chuỗi như là kiểu cơ bản (Java, C#, …)
- Trong C/C++, kiểu chuỗi không là kiểu cơ bản nhưng là kiểu được hỗ trợ
- Chuỗi có thể xem là mảng kí tự với hai dạng:
  - Lưu trữ tường minh độ dài chuỗi ở đầu chuỗi (Pascal)
  - Kết thúc chuỗi bằng kí tự đặc biệt (C/C++)

Kiểu chuỗi

- Trong C/C++, chuỗi là mảng kí tự (char) kết thúc bởi kí tự null ('\0')

```
char m1[] = {'h', 'e', 'l', 'l', 'o', '\0'};
char *p = m1;
m1++; // error
p[1] = 'E'; //m1[1]='E'
printf("%s", p + 1); //Ello
p = "hello";
p[1] = 'E'; // error
p++;
printf("%s", p); //ello
```

Kiểu hàm
Con trỏ hàm

- Kiểu hàm là kiểu cho phép quản lý, thao tác trên các hàm
  - Hàm cũng là dữ liệu
- Trong C/C++ kiểu hàm được thực thi bằng con trỏ hàm:
  - Kiến trúc máy Von-Neuman cho phép xem chương trình như là dữ liệu: chương trình cũng được lưu trong vùng nhớ như là dữ liệu
  - Vậy thì cũng có thể tham chiếu đến các hàm bằng con trỏ, gọi là con trỏ hàm

Kiểu hàm
Con trỏ hàm


```
int f1(int x) { return x; }
int f2(int x) { return 2 * x; }
void graph(int (*f)(int), int x1, int x2)
{
    for(int x = x1; x <= x2; x++)
          printf("(%d, %d) ", x, f(x));
}
void main()
{
    graph(f1, 1, 5);
    printf("\n");
    graph(f2, 1, 5);
}
```

Con trỏ và cấp phát động
Vùng nhớ cục bộ

- Vùng nhớ cục bộ (Stack)
  - là vùng nhớ riêng cho từng hàm
  - được tạo ra khi hàm bắt đầu thực thi
  - bị hủy khi hàm kết thúc thực thi
  - được quản lý tự động

```
int* f()
{ int a = 10; return &a; }
void main()
{
    int *p = f(); //vùng nhớ của a trong f đã bị hủy khi f kết thúc
    printf("%d", *p); //!!!
}
```

Con trỏ và cấp phát động
Vùng nhớ động

- Vùng nhớ động (Heap)
  - là vùng nhớ toàn cục (chung cho chương trình)
  - được tạo ra khi chương trình bắt đầu chạy
  - được giải phóng khi chương trình kết thúc
  - do lập trình viên quản lý theo cơ chế "xin – cho – dùng – hủy": `malloc/free` trong C, `new/delete` trong C++
  - lập trình viên truy cập đến dữ liệu trên vùng nhớ động thông qua con trỏ

Con trỏ và cấp phát động


```
int* f()
{
  int *p = new int; //p trỏ đến vùng nhớ trên Heap
  *p = 10;
  return p;
}
void main()
{
  int *p = f();
  (*p)++; // hợp lệ, vùng nhớ trên Heap dùng chung
  printf("%d", *p); //*p=11
  delete p;
}
```

Các lỗi thường gặp với con trỏ

- Chưa trỏ (dangling pointer)

```
int *p;
*p = 10; //!!!
```

- Tràn vùng đệm (buffer overflow)

```
int a = 10;
int b = 20;
int *p = &b;
printf("%d", *(p - 1)); //!!!
p = &a;
printf("%d", *(p - 1)); //!!!
```

Các lỗi thường gặp với con trỏ

- Trỏ bậy

```
int *p = new int;
*p = 10;
delete p;
printf("%d", *p); //!!!
delete p; //!!!
```

- Trỏ bậy

```
int* f1() { int a = 10; return &a; }
int* f2() { int *p = new int; *p = 10; return p;}
void main()
{
       printf("%d", *(f1())); //!!!
       printf("%d", *(f2())); //10
}
```

25

Hỏi và Đáp